STOCK MARKET PREDICTION USING LINEAR REGRESSION

723721243030 : LINGESHWARAN J

Phase 2 Submission Document

Project: Stock Market Prediction

Introduction:

Stock market prediction is a challenging and highly sought-after field within the realm of finance and data analytics. It involves using historical stock price and market data, along with various analytical and machine learning techniques, to forecast future movements in stock prices or market trends. Accurate stock market predictions are of great interest to investors, traders, financial institutions, and policymakers, as they can influence investment decisions, risk management, and overall market stability.

Here's an introduction to stock market prediction:

- The stock market is a critical component of the global financial system, where shares of publicly traded companies are bought and sold. It's characterized by volatility and influenced by a multitude of factors, including economic indicators, corporate performance, geopolitical events, and investor sentiment.

- Accurate predictions can help investors and traders make informed decisions about buying, selling, or holding stocks, potentially maximizing returns and minimizing losses.
- Financial institutions, including banks and investment firms, rely on stock market predictions to optimize their portfolio management and investment strategies.
- Governments and regulatory bodies may use these predictions to monitor market stability and implement policies to mitigate systemic risk.

Content for Project Phase 2 :

Innovating stock price prediction by exploring regression techniques like Linear Regression for improved Prediction accuracy.

1. Fundamental Analysis is the process of forecasting a company's future profitability based on its current business environment and financial performance.

2. Technical analysis, on the other hand, entails reading charts and analyzing statistical data to identify stock market trends. Here we'll concentrate on the technical analysis. To build a model capable of estimating stock prices, we will use the dataset of Microsoft stock prices from 1986 to 2020.

Data Source

A good data source for prediction using deep learning should be Accurate, Complete, Covering the geographic area of interest, Accessible.

Dataset Link:
(**https://www.kaggle.com/datasets/prasoonkottarathil/microsoft-lifetime-stocks-dataset**)

The dataset contains several variables, including date, open, high, low, close, and volume. The columns Open and Close represent the opening and closing prices of the stock on a given day. The maximum and minimum share prices for the day are represented by High and Low. The number of shares purchased or sold during the day is referred to as volume. Another thing to keep in mind is that the market is closed on weekends and public holidays.

Date	Open	High	Low	Close	Adj Close	Volume
13-03-1986	0.088542	0.101563	0.088542	0.097222	0.062549	1031788800
14-03-1986	0.097222	0.102431	0.097222	0.100694	0.064783	308160000
17-03-1986	0.100694	0.103299	0.100694	0.102431	0.065899	133171200
18-03-1986	0.102431	0.103299	0.098958	0.099826	0.064224	67766400
19-03-1986	0.099826	0.100694	0.097222	0.09809	0.063107	47894400
20-03-1986	0.09809	0.09809	0.094618	0.095486	0.061432	58435200
21-03-1986	0.095486	0.097222	0.091146	0.092882	0.059756	59990400
24-03-1986	0.092882	0.092882	0.08941	0.090278	0.058081	65289600
25-03-1986	0.090278	0.092014	0.08941	0.092014	0.059198	32083200
26-03-1986	0.092014	0.095486	0.091146	0.094618	0.060873	22752000
27-03-1986	0.094618	0.096354	0.094618	0.096354	0.06199	16848000
31-03-1986	0.096354	0.096354	0.09375	0.095486	0.061432	12873600
01-04-1986	0.095486	0.095486	0.094618	0.094618	0.060873	11088000
02-04-1986	0.094618	0.097222	0.094618	0.095486	0.061432	27014400
03-04-1986	0.096354	0.098958	0.096354	0.096354	0.06199	23040000
04-04-1986	0.096354	0.097222	0.096354	0.096354	0.06199	26582400
07-04-1986	0.096354	0.097222	0.092882	0.094618	0.060873	16560000
08-04-1986	0.094618	0.097222	0.094618	0.095486	0.061432	10252800
09-04-1986	0.095486	0.09809	0.095486	0.097222	0.062549	12153600
10-04-1986	0.097222	0.098958	0.095486	0.09809	0.063107	13881600
11-04-1986	0.098958	0.101563	0.098958	0.099826	0.064224	17222400
14-04-1986	0.099826	0.101563	0.099826	0.100694	0.064783	12153600
15-04-1986	0.100694	0.100694	0.097222	0.100694	0.064783	9302400
16-04-1986	0.100694	0.105035	0.099826	0.104167	0.067016	31910400
17-04-1986	0.104167	0.105035	0.104167	0.105035	0.067575	22003200
18-04-1986	0.105035	0.105035	0.100694	0.101563	0.065341	21628800

Data Collection and Preprocessing:

- Importing the dataset: Obtain a comprehensive dataset containing relevant features such as etc.
- Data preprocessing: Clean the data by handling missing values, outliers, and categorical variables. Standardize or normalize numerical features.
- The date column has been formatted as per the coding requirement.

Exploratory Data Analysis (EDA):

- Visualize and analyze the dataset to gain insights into the relationships between variables.
- Identify correlations and patterns that can inform feature selection and engineering.
- Present various data visualizations to gain insights into the dataset.
- Explore correlations between features and the target variable (Stock market prediction).

Inovation:

Innovating stock market prediction using linear regression is a challenging endeavor due to the inherent complexity of financial markets. While linear regression is a simple and interpretable method, innovating in this space involves employing it in novel ways and enhancing its capabilities

Program:

Importing required packages

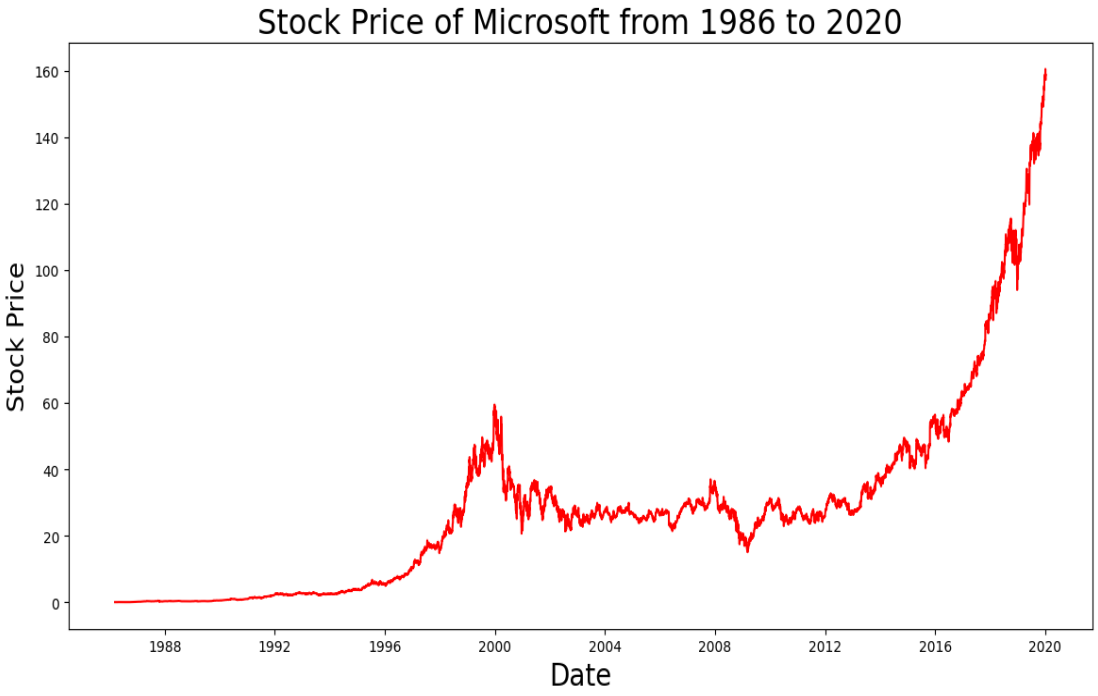
```
import numpy as np
import pandas as pd
import os
import matplotlib.pyplot as plt
import datetime as dt
from sklearn.linear_model import LinearRegression
```

Importing required Dataset

```
df =
pd.read_csv("C:/Users/KUMARAGURU/OneDrive/Documents/MSFT.csv")
df['Date'] = pd.to_datetime(df.Date,format='%m/%d/%Y %H:%M:%S')
df.index = df['Date']
```

Plotting the Data

```
plt.figure(figsize=(14,7))
plt.plot(df['Close'], label='Close Price history',color='r')
plt.xlabel('Date',size=20)
plt.ylabel('Stock Price',size=18)
plt.title('Stock Price of Microsoft from 1986 to 2020',size=23)
```



Spiting the Dataset

```
Shape=df.shape[0]
df_new=df[['Close']]
df_new.head()
train_data_set=df_new.iloc[:ceil(Shape*0.75)]
valid_data_set=df_new.iloc[ceil(Shape*0.75):]
```

```
    print("**********STOCK PRICE PREDICTION BY LINEAR
REGRESSION******")
    print('Shape of Training dataset Set',train_data_set.shape)
    print('Shape of Validation dataset Set',valid_data_set.shape)
```

Creating training and testing dataset

```
    train=train_data_set.reset_index()
    valid=valid_data_set.reset_index()
    x_train = train['Date'].map(dt.datetime.toordinal)
    y_train = train[['Close']]
    x_valid = valid['Date'].map(dt.datetime.toordinal)
    y_valid = valid[['Close']]
```

Implementing linear regression

```
    Model = LinearRegression()
    Model.fit(np.array(x_train).reshape(-1,1),y_train)
    preds = Model.predict(np.array(x_valid).reshape(-1,1))
    RMS=np.sqrt(np.mean(np.power((np.array(valid_data_set['Close'])-
preds),2)))
    print('(Root Mean Square Error)RMSE value on validation set:',RMS)
    valid_data_set['Predictions'] = preds
```

Plotting the Regression model

```
    plt.plot(train_data_set['Close'])
    plt.plot(valid_data_set[['Close', 'Predictions']])
    plt.xlabel('Date',size=18)
    plt.ylabel('Microsoft Stock Price',size=18)
    plt.title('Microsoft Stock Price Prediction by Linear Regression',size=18)
    plt.legend(['Model Training Data','Actual Data','Predicted Data'])
```

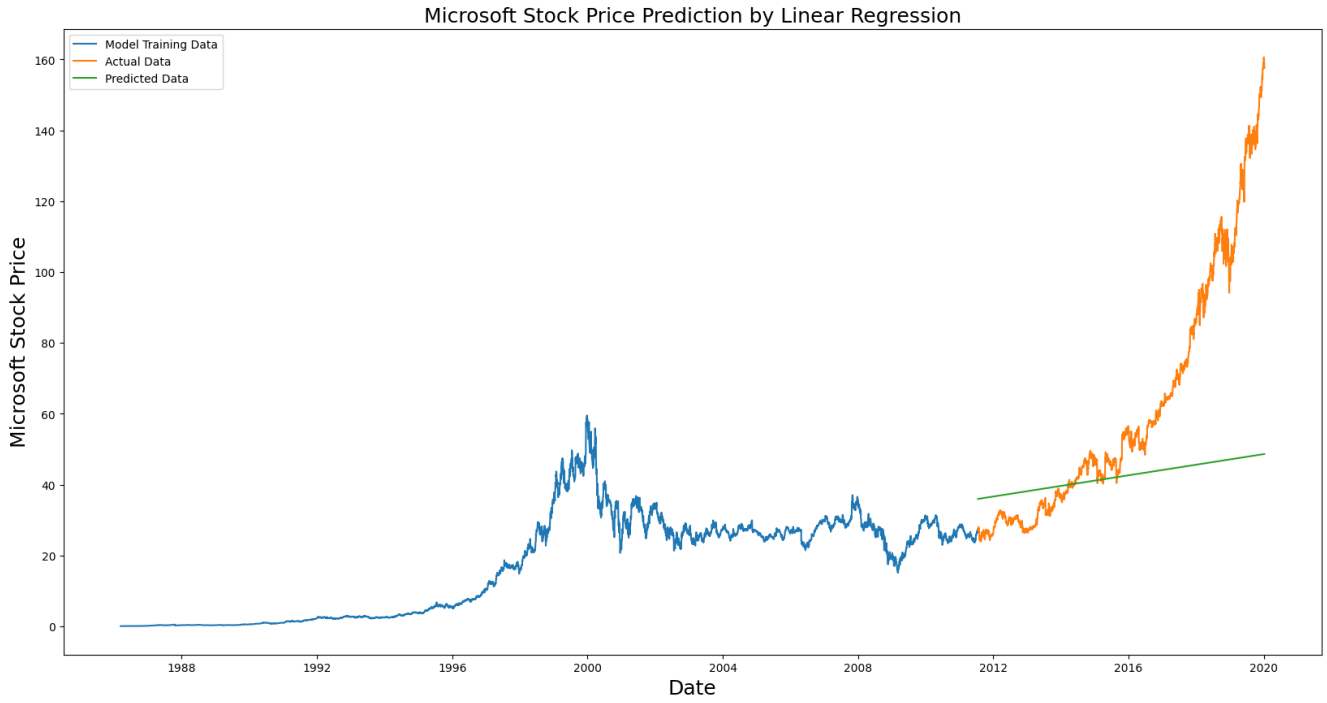
Output:

**********STOCK PRICE PREDICTION BY LINEAR REGRESSION******

Shape of Training dataset Set (6394, 1)

Shape of Validation dataset Set (2131, 1)

(Root Mean Square Error)RMSE value on validation set: 39.7286430309166



Conclusion:

Innovation in stock market prediction with linear regression offers a unique perspective on a well-established technique. Our exploration has revealed that while linear regression might not be the primary choice for predicting stock market trends, it can serve as a valuable complement to more sophisticated models.

By integrating linear regression with advanced feature engineering, data preprocessing, and external data sources, we can capture previously unnoticed relationships and anomalies in the market. This approach promotes a better understanding of market dynamics and can be particularly useful for long-term trend analysis and risk assessment. In conclusion, innovating stock market prediction with linear regression requires a creative approach, and while it may not replace complex models, it can provide valuable insights and enhance overall predictive accuracy.

Future scope :

The future of predicting Microsoft's stock price using LSTM presents a promising landscape. The proliferation of diverse data sources, including social media sentiment and economic indicators, will enable the creation of more comprehensive input features.

Advanced hardware, including quantum computing, will empower the handling of vast datasets with intricate LSTM architectures. The quest for model interpretability and explaining ability will continue, while reinforcement learning for optimized trading strategies will provide a more holistic approach to stock market prediction, positioning this application as a dynamic and influential domain in financial forecasting.